El proceso de descubrimiento de conocimiento en bases de datos

José Fernando Reyes Saldaña, Rodolfo García Flores
Posgrado en Ingeniería de Sistemas FIME-UANL
rodolfo@yalma.fime.uanl.mx
fernando@yalma.fime.uanl.mx

RESUMEN

La determinación de los patrones de compra es una fuente de información muy importante para el desarrollo de las estrategias de venta y compra de artículos en una empresa, que permitan satisfacer las necesidades de sus clientes. Para llegar a estos patrones se utiliza el proceso de "descubrimiento de conocimiento en bases de datos", el cual consiste de una serie de pasos que permite a identificar patrones poco obvios dentro de los datos. Este artículo describe el proceso del "descubrimiento de conocimiento en bases de datos" y algunas de sus aplicaciones actuales, e ilustra el proceso mediante el estudio de un caso a partir de una base de datos de clientes de una pequeña industria química.

PALABRAS CLAVE

Patrones de compra, descubrimiento de conocimiento en bases de datos, minería de datos, inteligencia artificial.

ABSTRACT

The discovery of buying patterns is a very important knowledge source for the development of selling and buying strategies in a company. Obtaining these patterns in a quick and easy way enables us to know and analiyse the needs of our clients, and learn what we can do to solve these needs. In order to extract these patterns we use the process of knowledge discovery in databases (KDD). This process consists of several steps that lead us to discover interesting patterns in our data. This paper describes the concept of knowledge discovery in databases and some of its current applications. We show this process through a case study using the database of a small chemical firm.

KEYWORDS

Buying patterns, knowledge discovery in databases, data mining, artificial intelligence.

INTRODUCCIÓN

El objetivo de este artículo es presentar el proceso de descubrimiento de conocimiento en bases de datos a través del análisis de la base de datos de una empresa cuyo nombre se omite por razones de confidencialidad. Esta empresa se dedica a la comercialización de productos químicos especializados. El realizar un

análisis de los datos de ventas proveerá información importante acerca de los hábitos de compra de sus clientes.

Para llegar a conocer los patrones existentes dentro de la base de datos se debe resolver un problema de *asociación*. Este tipo de problema se caracteriza por buscar patrones dentro de los datos para llegar a reglas que asocien los diferentes atributos de ellas. Para resolver el problema de interés para este artículo se analizará la información contenida en la base de datos en forma de *transacciones*, donde una transacción contiene los datos de los artículos comprados por un mismo cliente. Este problema se conoce como el *problema del carrito del supermercado*. Su propósito es estudiar los artículos adquiridos por un cliente para identificar combinaciones que tienen afinidad unos con otros, es decir, se trata de identificar la relación entre dos artículos presentes en la misma transacción. Por ejemplo, se espera ver en un supermercado que un cliente que ha comprado carne para asar, lleve en el mismo carrito el carbón, cebolla y todo lo necesario para asar la carne.

Sin embargo, se requerirá una gran cantidad de información sin ningún orden específico, ya que los clientes no suelen acomodarse según lo que compran. El trabajo del analista será el buscar de entre todos estos datos, cuáles pueden proveer información valiosa acerca de los hábitos de compra de los clientes. Para resolver este problema se utilizarán el *proceso de descubrimiento de conocimiento de bases de datos* (Knowledge Discovery in Databases, KDD por sus siglas en inglés) y la *minería de datos* (data mining), DM los cuales son muy estudiados en la actualidad debido a su amplia aplicación en las bases de datos corporativas, las cuales tienden a ser de gran tamaño.

García-Flores[1] menciona algunas de las aplicaciones de KDD, el cual se emplea en una gran cantidad de actividades, tales como:

Mercadeo: Ésta ha sido un área de aplicación tradicional de las técnicas de descubrimiento de conocimiento. La aplicación dentro del mercadeo está principalmente encaminada al análisis de las bases de datos de clientes. Por ejemplo, Fitzgerald[2] y Whitung[3] presentan cómo mejorar el proceso de venta de empresas mediante la minería de datos.

Inversiones financieras: Muchas aplicaciones de análisis financiero emplean técnicas de predicción para tareas como la creación de la cartera de clientes y la creación de modelos financieros, pero para mantener su ventaja competitiva, raramente se publican estos trabajos. Becerra[4] presenta un estudio del riesgo de inversión en diferentes países.

Detección de fraudes: Los bancos y otras instituciones financieras utilizan KDD para la detección de transacciones sospechosas y actividades de lavado de dinero. Sangi[5] realizó un estudio de transacciones sospechosas para la detección de fraudes bancarios.

Manufactura y producción: El KDD en planeación y control de manufactura es un área con gran potencial de ganancia, dado que los datos obtenidos son raramente explotados. Ho[6] muestra una aplicación de minería de datos en monitoreo y diagnóstico de manufactura remota.

Administración de redes: Esta área tiene un factor de cambio muy rápido con respecto al tiempo. Las redes de computadoras y telecomunicaciones son grandes y complejas y producen muchas alertas diariamente, pero también producen datos de los cuales se puede extraer conocimiento acerca de su operación. Mannion[7] presenta un producto que integra minería de datos para la administración de redes computacionales.

Minería de datos en Internet (Web mining): Es un área en auge debido al crecimiento exponencial de la Internet. Útil por ejemplo para el descubrimiento de patrones de navegación de los usuarios y para mejorar el diseño y organización de un sitio de Internet de

acuerdo a los patrones de acceso. Chang[8] presenta una aplicación de minería de datos en Internet para búsqueda de imágenes.

El impacto de KDD y DM para las empresas puede ser muy amplio, ya que su utilidad potencial depende de los resultados del análisis de datos. Pequeñas variaciones en los valores de los parámetros pueden producir resultados muy generales o demasiado particulares. Sin embargo, si se definen bien cuáles serán los datos de entrada y los datos de salida, se puede acotar el campo de estudio del análisis y definir el alcance del resultado que se espera obtener al utilizar KDD.

PROCEDIMIENTO PARA LA BÚSQUEDA DE INFORMACIÓN EN BASES DE DATOS

El proceso de KDD consiste de varios pasos, a través de los cuales se creará un modelo para el análisis de la base de datos. Estos pasos son:

1. *Aprender el dominio de la aplicación*. Implica el adquirir conocimiento del área de estudio del sistema y la meta a obtener. Se puede descomponer esta tarea en tres áreas:
   a. *Aprendizaje del tema*. El analista debe conocer el proceso detrás de la generación de la información para poder formular las preguntas correctas, seleccionar las variables relevantes a cada pregunta, interpretar los resultados y sugerir el curso de acción después de concluido el análisis.
   b. *Recolección de datos*. El analista debe conocer dónde se encuentran los datos correctos, cómo fueron obtenidos los datos de varias fuentes, cómo se pueden combinar estos datos y el grado de confianza de cada fuente.
   c. *Experiencia en análisis de datos*. El experto en DM debe tener conocimientos adecuados en el uso de la estadística.
2. *Creación de la base de datos de trabajo*. Consiste en elegir un subconjunto de variables o datos de muestra, de los cuales se obtendrá conocimiento. Esto con el fin de eliminar valores redundantes e inconsistencias en los datos de varias fuentes al juntarlos dentro de una sola base de datos.
3. *Limpieza y pre-procesamiento de los datos*. Incluye operaciones básicas sobre los datos, como el filtrado para reducir ruido y decidir qué hacer con los datos faltantes. Otras tareas de pre-procesamiento no tan evidentes son:
   a. *Derivar nuevos atributos*. Crear campos explícitos con relaciones entre los atributos conocidos (como relaciones entre ingresos y gastos) pueden hacer el análisis más sencillo.
   b. *Agrupación*. Donde hay relaciones uno-a-muchos en las bases de datos, podemos convertir estas relaciones en uno-a-uno y agregar un campo de conteo o suma, que contabilice todos los registros de la relación.
4. *Reducción de datos y proyección*. En este paso el analista trata de buscar características útiles para representar los datos en función de las metas del proyecto y posiblemente también reducir las dimensiones de la base de datos.
5. *Elegir la función del algoritmo de minería de datos*. El propósito del modelo se decidirá en este paso. Usualmente los algoritmos de DM realizan una de las siguientes tareas:
   a. *Síntesis*. Dados una gran cantidad de atributos, es necesario sintetizar los datos usando varias reglas características que simplificarán la construcción del modelo.
   b. *Asociación*. Los algoritmos en esta clase generan reglas que asocian patrones de transacciones con cierta probabilidad.
   c. *Agrupamiento*. Agrupar objetos dentro de clases, basados en sus características, maximizando la semejanza dentro de la misma clase, y minimizando la semejanza entre clases diferentes.
   d. *Clasificación y predicción*. Categorizar datos basándose en un conjunto de datos de entrenamiento y hacer un modelo para cada clase. Este modelo sirve para clasificar los nuevos datos agregados a la base de datos.
6. *Elegir el algoritmo de minería de datos*. La tarea consiste en seleccionar el método a ser usado para la búsqueda de patrones en los datos. Esto refina el alcance de la tarea anterior para utilizar el algoritmo más adecuado que ayude a alcanzar el objetivo final.

7. *Minería de datos*. Es el paso de análisis propiamente dicho.
8. *Interpretación*. Consiste en entender los resultados del análisis y sus implicaciones y puede llevar a regresar a alguno de los pasos anteriores. Hay técnicas de visualización que pueden ser útiles en este paso para facilitar el entendimiento.
9. *Utilización del conocimiento obtenido*. La aplicación de los patrones extraídos puede implicar uno de los siguientes objetivos:
   a. *Descripción*. La meta es simplemente obtener una descripción del sistema bajo estudio.
   b. *Predicción*. Las relaciones obtenidas son usadas para realizar predicciones de situaciones fuera de la base de datos.
   c. *Intervención*. Los resultados pueden conducir a una intervención activa en el sistema modelado.

El proceso puede contener varias iteraciones o ciclos entre los pasos. El punto crucial de este procedimiento se encuentra en el algoritmo de análisis (paso 6), que provee de una forma inteligente y automática de obtener conocimiento útil a partir de los datos. El paso central del KDD, la minería de datos, es un método de análisis apropiado cuando partimos de una pregunta vaga con muchas relaciones posibles por evaluar, por ejemplo "¿Qué grupos de clientes tienden a comprar X?". Por otro lado, si la pregunta es específica, los métodos estadísticos clásicos resultan más adecuados para abordar el estudio.

En la siguiente sección se presentan las herramientas que se utilizarán para el análisis de los datos con KDD. En las secciones restantes se ilustra la aplicación del proceso KDD al análisis de la base de datos de una pequeña empresa química.

HERRAMIENTAS DE ANÁLISIS

El descubrimiento de conocimiento se realizó a través de un programa en lenguaje Java. Se eligió este lenguaje debido a que es portátil, es decir, se puede utilizar en cualquier sistema operativo sin cambios en el programa original; está totalmente orientado a objetos, además de tener a disposición la biblioteca de funciones de análisis Weka.

La biblioteca de análisis Weka fue desarrollada por la Universidad de Waikato, Nueva Zelanda, y contiene un conjunto de algoritmos de aprendizaje de máquina. El utilizar esta biblioteca de análisis numérico permite centrarse más en el manejo de los datos y el formato de los resultados que en detalles de implementación de los algoritmos. Para poder procesar los datos, es necesario convertirlos a un formato de archivo especial, llamado ARFF.

A continuación se ilustrarán los pasos del KDD mediante el caso de estudio ya mencionado.

LA INFORMACIÓN A ANALIZAR

El primer paso del proceso de KDD es familiarizarse con el dominio de la aplicación y la meta a obtener. La base de datos de la empresa contiene información acerca de todos los movimientos realizados por el departamento de ventas durante un período de doce meses, los cuales totalizan trece mil seiscientos noventa movimientos. Cada entrada en esta base de datos representa una compra, como se puede ver en la figura 1. La meta del análisis es conocer qué artículos compran en común los clientes, es decir, si un cliente adquiere el artículo *A*, es posible que también adquiera el artículo *C*, en la misma compra o compras diferentes.

Como segundo paso se debe crear la base de datos de trabajo. Este proceso puede ser el más complicado, ya que si no tenemos bien definido el objetivo, cualquier subconjunto de datos puede parecer útil. Sin embargo, una vez que se sabe cuál es el resultado que se desea obtener, es posible definir más fácilmente qué datos serán necesarios.

En el caso del estudio que se presenta existen muchos datos en la base de datos original que no serán útiles para el análisis. Por ejemplo, en la figura 1, la columna con el número de cliente y su razón social representan la misma información, igualmente para el número de artículo y descripción. Debido a que se buscan los artículos comunes que compran los clientes, se considerará solamente el número de cuenta del cliente y el número de catálogo del artículo. El resto de los datos se descartará.

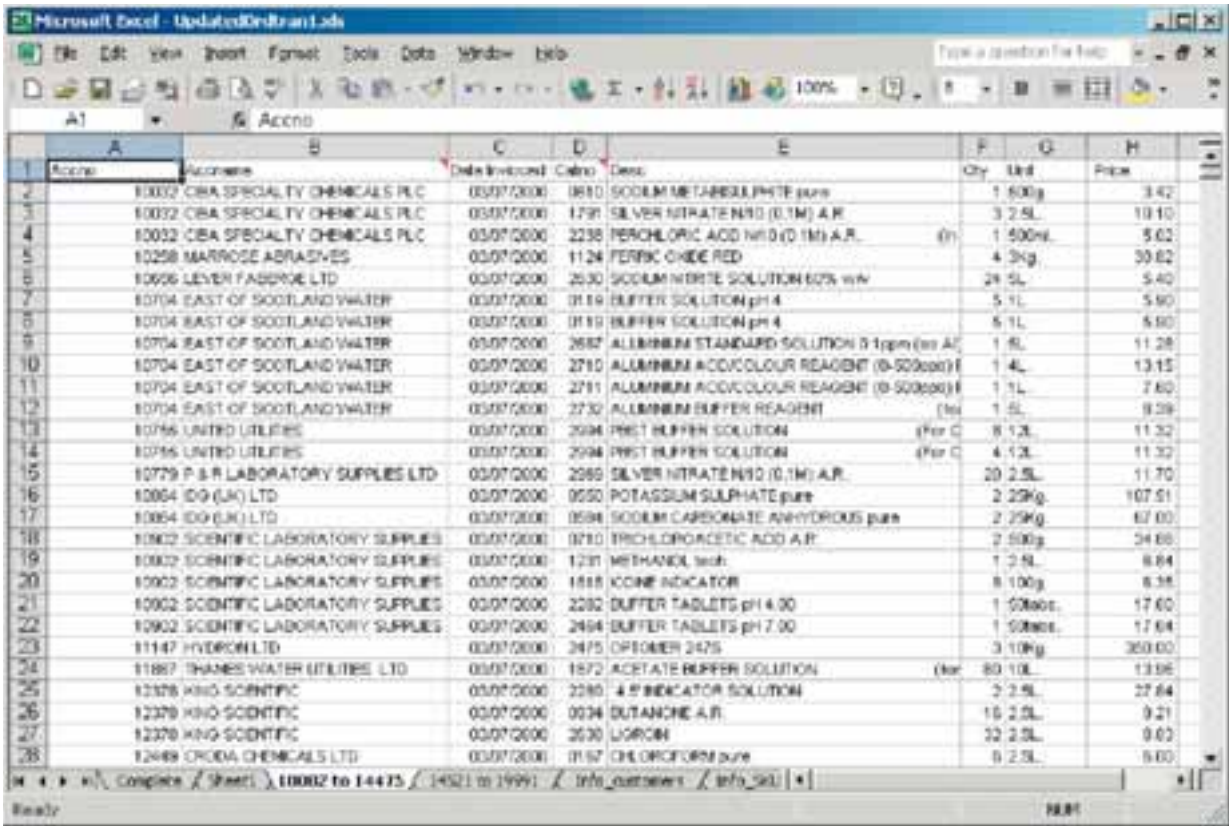

Fig. 1. Muestra de la base de datos de transacciones. Incluye número de cuenta de los compradores e información sobre el producto adquirido.

LIMPIEZA Y PRE-PROCESAMIENTO DE DATOS

El primer paso para la limpieza será el eliminar de la base de datos de trabajo productos comprados más de una vez por el mismo cliente, ya que como contiene los movimientos realizados por los clientes durante un período de doce meses, es de esperarse que los clientes hayan comprado un mismo artículo más de una vez en este período. Así, si se ordenan los artículos por número de cliente y número de artículo se pueden identificar grupos de cliente-artículo repetidos, que se pueden eliminar fácilmente.

Una vez ordenados los datos hay dos acciones que se deben realizar con ellos. La primera es obtener la lista de todos los artículos diferentes. La segunda, la eliminación de los productos repetidos, con el fin de preparar el archivo ARFF. Ambas tareas se realizarán mediante macros en Excel, debido a que se tienen una cantidad pequeña de datos en el caso de trabajo.

La eliminación de artículos repetidos reduce los movimientos de 13,690 a 3,725.

Una vez obtenida la lista de clientes y artículos, se acomodan todos los artículos comprados por un cliente en un renglón. Ya acomodados se obtiene una lista como la que se muestra en la figura 2.

Esta lista contiene los artículos en la primera columna y la lista de artículos por cliente en los renglones a partir de la columna D, como se muestra en la figura 2. A partir de este archivo se obtendrá el listado de transacciones para el archivo ARFF.

Se desarrolla otra macro más, que realizará la exportación desde los datos, haciendo lo siguiente:

1. Dentro de la lista de artículos, se marcan con un "1" los que se encuentran presentes en nuestro arreglo de la derecha, y se dejan con el "?" los que no se encuentran. Sólo se marcan los que tienen más de un artículo, ya que no se puede obtener una relación con un solo artículo.
2. Una vez terminado, se exporta la columna de valores al archivo ARFF. Esta columna representa una transacción.

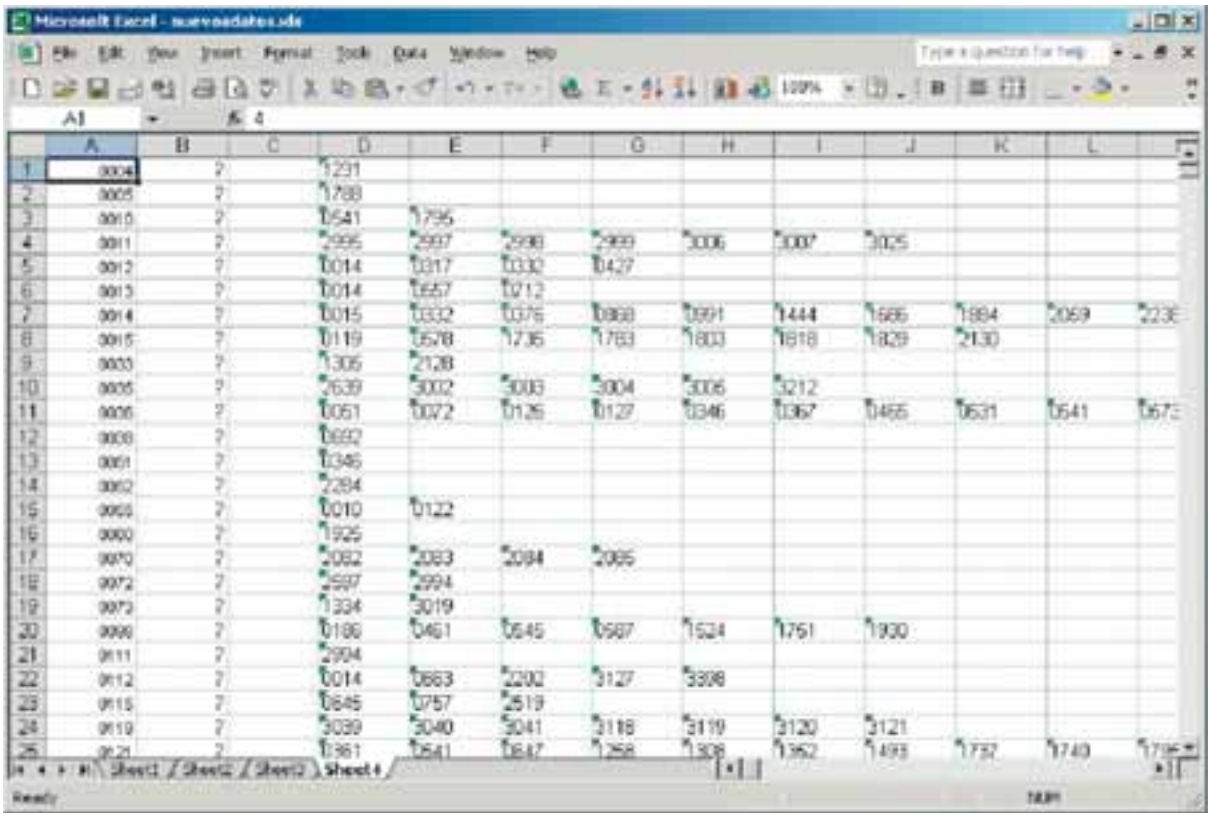

Fig. 2. Datos preparados para la exportación a la sección 3 del archivo ARFF. Se muestra el número de catálogo de los productos y los números de cuenta de los compradores.

3. Se reinician todas las celdas de valores a "?" y se prosigue con la siguiente línea.

Una vez terminado el archivo requerido por nuestro algoritmo, se elige como función de minería de datos la de asociación.

EL OBJETIVO DEL ANÁLISIS

Debido a que se espera obtener relaciones entre los diferentes productos que se encuentren dentro del conjunto de transacciones de la empresa, la función más apropiada para el análisis es el descubrimiento de reglas de asociación dada por Webb.[9]

El descubrimiento de reglas de asociación busca relaciones o afinidades entre conjuntos de artículos (item sets). Un conjunto de artículos se define como cualquier combinación formada por dos o más artículos diferentes de todos los artículos disponibles.

Una regla de asociación se forma con dos conjuntos: la premisa y la conclusión. La conclusión se restringe a un solo elemento. Las reglas generalmente se escriben con una flecha apuntando hacia la conclusión desde la premisa, por ejemplo $\{0041\} \rightarrow \{3465\}$. Una regla de asociación indica una afinidad entre la premisa y la conclusión, y generalmente está acompañada por estadísticos basados en frecuencia que describen esta relación.

Los dos estadísticos utilizados inicialmente para describir las relaciones son el *soporte* (o *apoyo*, denotado *sop*) y la *confianza* (*conf*), los cuales son valores numéricos. Para describirlos se necesitan algunas definiciones. Se define D como la base de datos de las transacciones, es decir, un conjunto de transacciones, y N como el número de transacciones en D. Cada transacción D_i es un conjunto de elementos, en el ejemplo un elemento es el número de artículo, como 0041 ó 3465. Se define $sop(X)$ como la proporción de transacciones que contienen el conjunto X, donde I es un conjunto de elementos, y se utilizará $|A|$ para denotar la cardinalidad del conjunto A.

$$sop(X) = \frac{\left|\{I \mid I \in D \wedge I \supseteq X\}\right|}{N} \quad (1)$$

El soporte de una regla de asociación es la proporción de transacciones que contienen tanto a la premisa como la conclusión. La confianza de una regla de asociación es la proporción de transacciones que contienen a la premisa, y que también contienen a la conclusión. Así, para una asociación $A \rightarrow C$:

$$sop(A \rightarrow C) = sop(A \cup C) \quad (2)$$

$$conf(A \rightarrow C) = \frac{sop(A \cap C)}{sop(A)} \quad (3)$$

A continuación se ilustra el cálculo del soporte con una pequeña base de datos de ejemplo que contiene 10 transacciones, mostrada en la figura 3. Se puede observar aquí que, si se quiere obtener *sop*(*manzanas*), de 10 transacciones disponibles 4 contienen *manzanas*, por lo que *sop*(*manzanas*) = 4/10 = 0.4*,* igualmente para el *sop*(*zanahoria*) hay 3 transacciones que la contienen, así *sop*(*zanahoria*) = 3/10 = 0.3*, sop*(*dulcería*) = 0.6*, sop*(*manzana→dulcería*) = 0.3*, sop*(*manzana→tomates*)=0.3.

{ciruelas, lechuga, tomates}
{apio, dulcería}
{dulcería}
{manzanas, zanahorias, tomates, papas, dulcería}
{manzanas, naranjas, lechugas, tomates, dulcería}
{duraznos, naranjas, apio, papas}
{frijoles, lechuga, tomates}
{naranjas, lechuga, zanahorias, tomates, dulcería}
{manzanas, plátanos, ciruelas, zanahorias, tomates, cebollas, dulcería}
{manzanas, papas}

Fig. 3. Base de datos de ejemplo.

Si el soporte o apoyo es suficientemente alto y la base de datos es grande, entonces la confianza es un estimado de la probabilidad que cualquier transacción futura que contenga la premisa, contendrá también la conclusión. De la base de datos de ejemplo de la figura 3, vemos que *conf*(*manzanas→dulcería*) = *sop*(*manzana→dulcería*) / *sop*(*manzanas*) = 0.3/0.4 = 0.75*, conf*(*manzanas→tomates*) = 0.75*, conf*(*zanahorias→dulcería*) = 1.

El algoritmo de asociación tratará de descubrir todas las reglas que excedan las cotas mínimas especificadas para el soporte y la confianza. La búsqueda exhaustiva de reglas de asociación consideraría simplemente todas las combinaciones posibles de elementos, poniéndolas como premisas y conclusiones, entonces se evaluaría el soporte y la confianza de cada regla, y se descartarían todas las asociaciones que no satisfacen las restricciones. Sin embargo el número de combinaciones crece rápidamente con el número de elementos, por lo que si hay 1,000 elementos, se tendrán $2^{1,000}$ combinaciones (aproximadamente 10^{300}). Para cada premisa existe la posibilidad de formar una regla poniendo como conclusión cualquier conjunto de elementos que no contenga algún elemento que ya se encuentra en la premisa. Así, este procedimiento para la búsqueda de reglas de asociación es muy costoso computacionalmente, por lo que se necesita otro procedimiento más eficiente.

EL ALGORITMO APRIORI

El algoritmo *Apriori* presentado por Agrawal[10] ataca el problema reduciendo el número de conjuntos considerados. El usuario define un soporte mínimo, min_sop. De la definición de soporte tenemos que si $sop(A \cup C) \leq \texttt{min\_sop}$ entonces $sop(A \rightarrow C) \leq \texttt{min\_sop}$. Apriori genera todos los conjuntos que cumplen con la condición de tener un soporte menor o igual a min_sop. Para cada conjunto frecuente X se generan todas las reglas de asociación $A \rightarrow C$ tales que $A \cup C = X$ y $A \cap C = \varnothing$. Cualquier regla que no satisfaga las restricciones impuestas por el usuario, como por ejemplo la confianza mínima, se desechan, y las reglas que sí cumplen se conservan.

Como $sop(A) \geq sop(A \rightarrow C)$ y $sop(C) \geq sop(A \rightarrow C)$, si $A \cup C$ es un conjunto frecuente entonces tanto A como C son conjuntos frecuentes. El soporte, la confianza, y otras métricas por las cuales la regla de asociación $A \rightarrow C$ es evaluada pueden ser derivadas desde $sop(A)$, $sop(C)$ y $sop(A \cup C)$. Así, guardando todos los conjuntos frecuentes y su soporte, tenemos toda la información requerida para generar y evaluar las

reglas de asociación que satisfacen min_sop.

En la solución del problema del carrito de supermercado, cada producto individual aparece solamente en una pequeña cantidad del total de las transacciones. Así, el número de conjuntos frecuentes es relativamente bajo, aún cuando min_sop sea un valor muy pequeño. Por eso, el utilizar conjuntos frecuentes nos permite reducir el espacio de búsqueda a un tamaño más manejable, debido a que los datos del carrito de compras se encuentran muy dispersos.

La búsqueda inicial de reglas de asociación permite encontrar todas las asociaciones que satisfagan una restricción inicial de soporte y confianza. Esto puede llevar a obtener una gran cantidad de reglas de asociación a partir de los datos, las cuales no serían manejables. Por lo tanto es deseable reducir el número de reglas de tal manera que solo queden las más interesantes. Para esto se utilizan otras medidas de interés de las reglas de asociación como el levantamiento y el apalancamiento.

LEVANTAMIENTO

Esta medida compara un subconjunto de los datos contra todos los datos, dando resultados más generalizados que el soporte y la confianza, los cuales sólo nos proveen resultados evaluados en un subconjunto de los datos. El levantamiento (*lev*) se define como la relación entre la frecuencia con que la conclusión se encuentra en las transacciones que contienen la premisa, dividida entre la frecuencia de la conclusión en todos los datos.

$$lev(A \rightarrow C) = \frac{conf(A \rightarrow C)}{sop(C)} \qquad (4)$$

Valores de levantamiento mayores a 1 indican que la conclusión es más frecuente en transacciones que contienen también la premisa, que en transacciones que no la contienen.

Por ejemplo, considerando la asociación $\{tomates\} \rightarrow \{lechuga\}$. Si $sop(\{lechuga\})$=0.4 y
$conf(\{tomates\} \rightarrow \{lechuga\})$=0.67.
Entonces,

$$lev(\{tomates\} \rightarrow \{lechuga\}) = \frac{conf(tomates \rightarrow lechuga)}{sop(lechuga)}$$

$$= \frac{0.67}{0.4} = 1.675$$

Como contraste, consideramos otra asociación con la misma confianza,

$\{tomates\} \rightarrow \{dulcería\}$. Donde $sop(\{dulcería\})$ = 0.6, $conf(\{tomates\} \rightarrow \{dulcería\})$ = 0.67. Así,

$$lev(tomates \rightarrow dulcería) = \frac{conf(tomates \rightarrow dulcería)}{sop(dulcería)}$$

$$= \frac{0.67}{0.6} = 1.117$$

Estos valores relativos de levantamiento indican que los tomates tienen un mayor impacto en la frecuencia de la lechuga que en la frecuencia de la dulcería.

APALANCAMIENTO

Aunque el levantamiento es muy usado, no es siempre una buena medida de qué tan interesante puede ser una regla. Una asociación con poca frecuencia y mucho levantamiento puede ser de menor interés que una de mucha frecuencia pero poco levantamiento, debido a que esta última aplica a más individuos.

El apalancamiento (*ap*) es una medida que captura tanto el volumen como la fuerza de la regla en un sólo valor, y se define como la diferencia entre la frecuencia con la que la premisa y la conclusión ocurren y la frecuencia que se esperaría si ambos fueran independientes.

$$ap(A \rightarrow C) = sop(A \rightarrow C) - sop(A) \cdot sop(C) \qquad (5)$$

Por ejemplo, considérense las asociaciones $\{zanahorias\} \rightarrow \{tomates\}$ y $\{lechuga\} \rightarrow \{tomates\}$. Ambas asociaciones tienen confianza = 1.0 y levantamiento = 1.667. Aunque el segundo puede ser de mayor interés por aplicar a más clientes.

Podemos constatar esto al calcular el apalancamiento para $\{zanahorias\} \rightarrow \{tomates\}$ así,

$$sop(\{zanahorias\} \rightarrow \{tomates\}) = 0.3$$
$$sop(\{zanahorias\}) = 0.3$$
$$sop(\{tomates\}) = 0.6 \quad \therefore$$
$$ap(\{zanahorias\} \rightarrow \{tomates\}) = 0.3 - 0.3 \cdot 0.6 = 0.12$$

Y el apalancamiento para $\{lechuga\} \rightarrow \{tomates\}$,

$$sop(\{lechuga\} \rightarrow \{tomates\}) = 0.4$$
$$sop(\{lechuga\}) = 0.4$$
$$sop(\{tomates\}) = 0.6 \quad \therefore$$
$$ap(\{lechuga\} \rightarrow \{tomates\}) = 0.4 - 0.4 \cdot 0.6 = 0.16$$

El impacto final de la segunda asociación es mayor que el de la primera.

Medidas como el levantamiento y apalancamiento pueden ser usadas para restringir el número de reglas que obtenemos con el descubrimiento de reglas de asociación, proponiendo un valor mínimo para que éstas sean descartadas y obtener las mejores.

Ahora que se conoce el procedimiento a utilizar para realizar la minería de datos y se han procesado los datos para su utilización en el algoritmo, se presenta enseguida el modo en que se resolvió el problema propuesto en el presente trabajo.

RESULTADOS OBTENIDOS

Una vez programado el algoritmo y listo para ser ejecutado en Java, es necesario proveer los parámetros adecuados para obtener una buena cantidad de reglas de asociación. Los parámetros provistos son:

- Soporte mínimo = 0.05: Es el soporte mínimo a tener para que la regla sea considerada. Este soporte es muy pequeño debido a la relación entre la cantidad de reglas y la cantidad de atributos que se tienen. Dado que, como ya lo dijimos anteriormente, la matriz de transacciones de un problema de carrito de compras es una matriz dispersa, necesitamos utilizar un valor de confianza muy bajo para obtener reglas desde nuestro archivo. Es por esto que dentro de nuestro algoritmo definimos el soporte mínimo en 0.05.
- Tipo de métrica = Confianza: Las opciones disponibles para esta opción son los cuatro tipos de métricas explicadas anteriormente: soporte, confianza, levantamiento y apalancamiento. En este caso se indica que se considerarán las reglas con la confianza indicada.
- Número de reglas = 20: Indica el número máximo de reglas a obtener. Se utiliza como criterio de parada para detener la ejecución si se llega a este número de reglas cumpliendo con las restricciones propuestas.

Una vez que se ha preparado el archivo de datos, y se ha implementado el algoritmo, éste se ejecuta y se obtiene una ventana de resultados como la mostrada en la figura 4.

Aquí se puede observar que el algoritmo se ha ejecutado y ha dado como resultado un conjunto de 6 reglas de asociación para los datos. Estas reglas

```
Instancias: 312
Atributos: 1028

Finds association rules.

Apriori
=======

Minimum support: 0.05
Minimum metric <confidence>: 0.2
Number of cycles performed: 19

Generated sets of large itemsets:

Size of set of large itemsets L(1): 22

Size of set of large itemsets L(2): 3

Best rules found:

 1. 1829=1 18 ==> 0122=1 16    conf:(0.89)
 2. 1829=1 18 ==> 0119=1 16    conf:(0.89)
 3. 0119=1 26 ==> 0122=1 21    conf:(0.81)
 4. 0122=1 30 ==> 0119=1 21    conf:(0.7)
 5. 0119=1 26 ==> 1829=1 16    conf:(0.62)
 6. 0122=1 30 ==> 1829=1 16    conf:(0.53)
```

Fig. 4. Ventana de resultados mostrada por el algoritmo de minería de datos, después de ejecutarlo con el archivo generado mediante los datos originales.

indican que cuando un cliente adquiere un artículo, adquiere también el otro.

Si se cambia la restricción de soporte mínimo, a 0.02, se obtiene el conjunto de reglas mostrado en la figura 5.

```
 1. 2626=1 15 ==> 2627=1 15    conf:(1)
 2. 2628=1 12 ==> 2626=1 2627=1 12    conf:(1)
 3. 2626=1 2628=1 12 ==> 2627=1 12    conf:(1)
 4. 2627=1 2628=1 12 ==> 2626=1 12    conf:(1)
 5. 2628=1 12 ==> 2627=1 12    conf:(1)
 6. 2628=1 12 ==> 2626=1 12    conf:(1)
 7. 2627=1 16 ==> 2626=1 15    conf:(0.94)
 8. 1829=1 18 ==> 0122=1 16    conf:(0.89)
 9. 1829=1 18 ==> 0119=1 16    conf:(0.89)
10. 0119=1 1829=1 16 ==> 0122=1 14    conf:(0.88)
          ...
          ...
30. 1803=1 27 ==> 1797=1 12    conf:(0.44)
31. 1803=1 27 ==> 1795=1 12    conf:(0.44)
32. 0541=1 27 ==> 1803=1 12    conf:(0.44)
33. 1803=1 27 ==> 0541=1 12    conf:(0.44)
34. 1803=1 27 ==> 0119=1 12    conf:(0.44)
35. 0557=1 30 ==> 0014=1 12    conf:(0.4)
36. 0014=1 40 ==> 0557=1 12    conf:(0.3)
```

Fig. 5. Listado de reglas de asociación obtenidas por el algoritmo de minería de datos después de cambiar el soporte mínimo, para obtener un mayor número de reglas.

Una vez obtenidos los resultados, es necesario interpretarlos. Para ello es útil conocer las situaciones externas que generaron los datos. Por ejemplo, algunas reglas relacionan los productos 2626, 2627 y 2628, que son soluciones búfer analíticas de pH 4, 7 y 10, respectivamente. Una regla adicional relaciona estos compuestos con el yoduro de potasio, que se usa también como reactivo analítico. Tiene sentido suponer que los laboratorios químicos se surten de todos sus reactivos analíticos con el mismo proveedor. Para interpretar otro conjunto de productos frecuentes, notamos que muchos de los clientes de la empresa en cuestión son escuelas, por lo que ácidos y bases fuertes tienden a aparecer juntos. Por ejemplo, se venden juntas soluciones de amoniaco, hidróxido de sodio, y ácidos clorhídrico y sulfúrico en diferentes concentraciones. Se encuentra en una combinación más interesante que algunas reglas asocian soluciones búfer de acetato y fosfato, que son usados en los laboratorios de las plantas de tratamiento de aguas. Aunque no todas estas explicaciones son igualmente útiles o interesantes para la empresa, la información obtenida definitivamente puede apoyar la toma de decisiones en planta o para tomar medidas relacionadas con el manejo de inventario.

Es ilustrativo tomar en cuenta el nivel de confianza obtenido con cada regla. En la figura 4 se puede observar que las primeras 2 reglas tienen un nivel de 0.89, lo que indica que en la mayor parte de las transacciones estas reglas son ciertas, sin embargo en un 0.11 de las reglas, no sucede así. Con la regla número 6, que solamente tiene un nivel de confianza de 0.53, por lo que esta regla se aplica a aproximadamente la mitad de las transacciones que contienen al artículo 0122.

COMENTARIOS FINALES

En el presente artículo se mostró la aplicación del proceso de KDD y su paso central, la minería de datos, a través de un caso de estudio, del cual se extrajo un conjunto de reglas de asociación mediante el algoritmo conocido como *Apriori*. Este conjunto de reglas permite realizar el análisis de los patrones de compra de productos por parte de los clientes. La minería de datos es apropiada cuando la pregunta inicial es vaga y hay que evaluar las muchas relaciones posibles entre los atributos, por ejemplo "¿Qué grupos de clientes tienden a comprar X?". En cambio, si la pregunta es más específica, los métodos estadísticos clásicos son los más adecuados para emprender el estudio.

Dada la naturaleza de la pregunta que plantea el problema del carrito de supermercado, el análisis de las reglas de asociación puede llevar a obtener resultados que de otra manera hubiera sido imposible conocer, ya que el análisis manual de los datos no es fácil, y el obtener reglas por medios empíricos, como la experiencia, es poco confiable.

Como se pudo observar, el tratamiento de los datos es un proceso muy laborioso y que puede tomar gran parte del tiempo utilizado en todo el proceso de KDD, ya que es muy importante el definir los datos de entrada para obtener resultados satisfactorios.

Los patrones obtenidos como resultado permiten

realizar un análisis de los productos en común que los clientes compran, y con esto obtener algunas aplicaciones prácticas como son mejores estrategias de compra y venta, de acomodo de productos, diseño de promociones, entre otras.

El proceso de KDD es una herramienta importante para el análisis de los patrones de compra de los clientes, que puede ayudar a las empresas a obtener una ventaja competitiva muy valiosa.

REFERENCIAS

1. García-Flores, R. A multi-agent system for chemical supply chain simulation, management and support. PhD tesis, University of Leeds, United Kingdom, 2002.
2. Fitzgerald, K., Grocery Cards get and Extra Scan, *Credit Card Management*, Marzo 2004, 34-49.
3. Withing, R. Making it easier to predict future, *Information Week*, 26 de Abril de 2004, Pg. 56.
4. Becerra-Fernandez, I.; Zanakis, S.H.; Steven Walczak. Knowledge discovery techniques for predicting country investment risk. *Computers and Industrial Engineering*, 2002, 43: 787-800.
5. Songini, M.L., Fraud Sniffers, *ComputerWorld*, 21 de Junio de 2004, Pg. 42.
6. Hou, T., Liu, W., Lin, L., Intelligent remote monitoring and diagnosis of manufacturing processes using an integrated approach of neural networks and rough sets, *Journal of Intelligent Manufacturing*, Abril 2003, Pg. 239.
7. Mannion, P. Vernier rethinks WLAN management software. *Electronic Engineering Times*, Manhasset, 2 de Febrero de 2004, número 1306, pg. 43.
8. Chang, Z., Wenyin, L., Zhang, F., Li, M.; Zhang, H. Web mining for web image retrieval, *Journal of the American Society for Information Science and Technology*, Agosto 2001, Pg. 831.
9. Webb, G. I., Association Rules. In the handbook of data mining, Ye, N. (Ed.), Laurence Erlbaum Publisers, Londres 2003, Pg. 25-39.
10. Agrawal, R., Srikant, R., Fast Algorithms for Mining Association Rules, *Proceedings of the 20th VLDB Conference,* IBM Almaden Research Center, 1994.